CHAPTER 5

Prognostic Performance Metrics

Kai Goebel
Abhinav Saxena
Sankalita Saha
Bhaskar Saha
Jose Celaya

CONTENTS

5.1 INTRODUCTION

We define prognostics here strictly as the predicting of remaining useful life (RUL) of a component or system. The prediction is typically performed only after the "health" of the component or system deteriorates beyond a certain threshold. Often times, that threshold is tripped because a fault occurs. A fault is a state of a component or system that deviates from the normal state such that the integrity of the component is outside of its required specification. A fault does not necessarily imply that the overall system does not operate anymore; however, the damage that characterizes the fault often grows under the influence of operations to a failure. The latter is the state at which the component or system does not meet its desired function anymore. It is the task of prognostics to estimate the time that it takes from the current time to the failed state, conditional on anticipated future usage. This would give operators access to information that has significant implications on system safety or cost of operations. Where safety is impacted, the ability to predict failure allows operators

to take action that preserves the assets either through rescue operation or through remedial action that avert failure altogether. Where minimizing cost of operations is the primary objective, predictive information allows operators to avert secondary damage or to perform maintenance in the most cost-effective fashion. Often times, there is a mix of objectives that need to be optimized together, sometimes weighted by different preferences.

Predicting remaining component or system life can be accomplished in several ways. Where sufficient historical run-to-failure trajectories are available, data mining techniques can be employed to perform the predictions. Traditionally, reliability-based predictions have been used widely in the manufacturing industry to schedule preventive maintenance. In contract, the focus of this chapter is mainly on condition-based prognostic systems for a particular monitored unit under test (UUT). Instead of considering the entire population for a statistical life estimates, one can employ physics-based models to perform the predictions or a combination of models and history data. In either case, predictions are conditional on future conditions and are subject to significant amounts of uncertainty. Methods for prognostics ideally express their confidence of their own prediction based on an assessment of the various uncertainty sources. Besides uncertainty of future usage, uncertainty also comes from the current state assessment, the models used, measurement noise, etc.

Metrics can be understood as a standardized language by which technology developers and users communicate their findings and compare results. This aids in allowing the proper expression of requirements as well as the dissemination of scientific information. Two surveys on methods for prognostics, one on data-driven methods [1] and one on artificial-intelligence–based methods [2] reveal a lack of standardized methodologies for performance evaluation or a lack of performance methods altogether. The most recent ISO standard by the International Organization for Standards [3] for prognostics in condition monitoring and diagnostics of machines does not even provide a firm definition of any such method. Nonetheless, there has been recently a significant push towards crafting suitable metrics to evaluate prognostic performance [4,5]. These metrics address primarily evaluation of algorithmic performance for prognostics applications. They are mostly focused on tackling offline performance evaluation methods for applications where run-to-failure data are available and true end-of-life (EoL) is known a priori. They are therefore particularly useful for the algorithm development phase where feedback from the metrics can

There are two Saxena et al. 2010. Please check which should be cited here.

be used to fine-tune prognostic algorithms. It needs to be appreciated that these metrics are continuously evolving. Efforts are also underway towards designing on-line performance metrics although they have not reached a significant level of maturity.

This chapter presents a discussion on prognostics metrics. After a review of performance assessment for prediction/forecasting applications a categorization of prognostic metrics into several classes is performed. This categorization suggests that there can be various different objectives that drive improvements in prognostic performance and correspondingly different set of metrics may be used to obtain performance feedback.

5.2 BACKGROUND

As more diverse research communities and practitioners start adopting prognostics and health management (PHM) techniques, it becomes imperative to use standardized prognostic methodologies [6] as well as to use metrics to measure performances. However, since prognostics is a developing field, the challenges in developing standards and metrics are numerous [7,8]. We start out by providing an overview of prognostic concepts that are used in a variety of domains.

5.2.1 Prediction Categorization

Prior to delineating the methods to assess prognostic performance it may be useful to provide a brief discussion about different types of applications in which predictions are employed. Based on an analysis of aerospace, medicine, nuclear energy, finance, weather, and automotive domains, it was found that one can distinguish roughly between forecasting and prognostics.

5.2.1.1 Forecasting

Forecasting is found in applications where predictions are made to describe expected future behavior without predicting a fixed target. That is, there is no notion of EoL, and consequently, there is no concept of RUL. Example application areas are weather or finance domains. The prediction format can be either quantitative (e.g., prediction of exact numbers) or qualitative (e.g., high or low demands) in nature. Furthermore, the data trends are generally nonmonotonic in such applications. Predictions may be discrete (e.g., forecasting market demand for a particular month) or continuous (e.g., variation of temperature over the period of next week).

Details and more references to such applications in various domains can be found in [29].

5.2.1.2 Prognostics

The other class of applications makes use of critical thresholds such that if the system under test crosses this threshold it is declared to have failed or lost its functional capability. This class of applications—e.g., medicine, nuclear, mechanical, and electrical industrial systems—involves predictions of RUL and involves decay or fault propagation models to capture the behavior of the system.

Predictions can be made in two forms: (1) an event prediction where the time for EoL is estimated and (2) a decay prediction where the complete future trajectory is predicted until EoL is reached. It must be noted, however, that EoL criteria need not always be a complete loss or failure. In safety critical applications, EoL is often a degraded state where performance level has deteriorated to cross a predetermined safety margin even though the component may still retain partial functionality. For example, in the electronics domain, EoL of a switching device (such as a MOSFET) is not necessarily the complete loss of the switching functionality. Instead, it could be a decrease in the switching frequency below a certain threshold level.

There are two main types of applications where predictions for system health are made: These include *predicting wear* of a system or *predicting failure* in the event of a fault.

Failure predictions: An otherwise healthy system may encounter a fault that grows due to continued usage (or exposure to adverse conditions) that may result into a failure. In such cases, it is critical to detect the presence of a fault (ideally shortly after it happens), the particular fault mode, its severity, and its rate of growth so that appropriate decisions may be taken to avoid undesired, possibly catastrophic, events. Here, the task of prognosis is to estimate expected EoL, i.e., determine when the system will no longer operate under specifications. In some cases, it is not only important to know when the system will break but also how it will approach the failure. In those cases, instead of predicting just the event of EoL, a complete trajectory may be predicted, where the end point of the trajectory also determines the EoL. Examples of such applications include structural faults such as cracks in metallic structures or die-attach degradation in power semiconductor devices.

Wear predictions: There are many situations where systems undergo expected normal wear and need to be maintained or replaced whenever the wear levels impact functionality. In these cases, the system does not experience a fault condition even under the degraded performance. Therefore, the health of the system is tracked from the very beginning of system deployment and detection and diagnosis are not predecessors for prognostics. As stated earlier, the end point of these trajectories can be used to determine the EoL point so appropriate decisions may be taken. Examples of such applications include battery capacity degradation and valve leakage due to wear.

5.2.2 Prediction Methods

There are several ways to carry out prognostics. In some cases, a detailed physical model of the unit under observation can be used. The model captures the unit's behavior under operational and environmental conditions and provides an expected response that describes the current and (given the proper input) future states. Alternative to a physics-based model, historical data can be utilized to estimate expected time to failure. The key is to either have access to a sufficient amount of existing historical data (e.g., medicine) or to be able to experimentally generate run-to-failure trajectories (e.g., for some mechanical systems). Then, a variety of data-driven or statistical techniques can be applied.

The availability of run-to-failure data allows the straightforward evaluation of prediction performance by comparing the predicted EoL to the actual EoL. However, there are many applications where run-to-failure experiments cannot be afforded or where very little failure history data are available (e.g., aerospace). It becomes somewhat more difficult to assess the performance in such cases due to the absence of knowledge about the future outcomes. Methods are tested on experimental or simulated data and, when fielded, are expected to perform similarly on real systems. However, algorithm functionality does rarely translate without loss of performance from simulation environment or laboratory to the field. Indeed, validation and verification of prognostic methods remain a thorny issue.

5.2.3 Performance Evaluation Methods

Techniques employed for prediction or forecasting in the application areas enumerated above use metrics that are based on accuracy and precision with several slight variations [9]. Mostly, they are customized to better serve a particular domain. In medicine and finance, for example, several

statistical measures are used to benefit from the availability of large data sets. In contrast, predictions in medicine are commonly evaluated based on hypothesis testing methodologies. In the finance domain, errors are calculated based on reference prediction models. The precision and accuracy metrics include, for example, mean squared error (MSE), standard deviation (SD), mean absolute deviation (MAD), median absolute deviation (MdAD), mean absolute percentage error (MAPE), and similar variants. Other domains, such as aerospace, electronics, and nuclear are less mature with respect to fielded prognostics applications. There, metrics from other system health techniques, such as diagnostics, have been used with the goal to capture the characteristics of prognostics (with varied success). Metrics used include false-positives (FP), false-negatives (FN), and receiver operator characteristics (ROC) curves [10]. Other metrics include those from the reliability domain such as mean time between failures (MTBF) or the ratio mean time between unit replacements (MTBF/MTBUR). Adaptations include, for example, the augmentation with business metrics such as return on investment (ROI) [11], technical value (TV) [12], net present value (NPV) [13], and life cycle cost (LCC) [14].

It becomes apparent that there are several types of metrics for prognostics based on the purpose of prognostics and the end user. A categorization with these objectives in mind allows a more targeted choice of appropriate metrics. Coble and Hines [15] categorized prognostic algorithms into three categories based on type of models/information used for predictions. Wheeler et al. [16] categorized end users from a health management stakeholder's point of view. The top-level user groups were operations, regulatory, and engineering. We combine and expand on these notions and categorize prognostic metrics based both on their goal as well as end users (see Table 5.1).

A favorable cost-benefit case is the hinge pin of a successfully fielded prognostic solution and cost-benefit metrics allow a quantification of the degree of fulfillment. Similarly, verifiability and certification metrics determine the degree to which a prognostic solution conforms with safety assurance and certification requirements. Both these top-level metrics categories require prognostic estimates to satisfy stringent performance metrics, which are often derived from reliability analysis or condition-based prognostic methods. Computational performance metrics are important for implementation and figure in the trade space of cost-benefit analysis and algorithmic performance. Often, most of the metrics mentioned are connected through a requirement specification process.

TABLE 5.1 Categorization of Prognostic Metrics Based on End Usage

		Operations				Engineering		Regulatory
Metrics	**Assessment Goals**	**Program Manager**	**Plant Manager**	**Operator**	**Maintainer**	**Designer**	**Researcher**	**Policy Maker**
Certification metrics	Assess conformance to safety assurance and certification requirements	X						X
Cost-benefit metrics	Assess the economic viability for specific applications before it can be approved or funded	X						X
Reliability-based metrics	Assess probabilities of failures based on statistical evidence from multiple systems			X		X		X
Algorithm performance metrics	Assess performance of prediction algorithms in predicting EoL	X	X	X	X		X	X
Computational performance metrics	Assess computational requirements					X	X	

5.3 METRICS FOR PROGNOSTIC APPLICATIONS

In this section, the various prognostic metrics are presented by categories in more detail. Particular emphasis is given to performance metrics.

5.3.1 Certification Metrics

Regulatory bodies (such as the FAA) are concerned with whether a fielded system might negatively impact (directly or indirectly) overall system safety. Associated metrics can be expressed as the logical conjunction of an exhaustive set of safety-related use cases.

5.3.2 Cost-Benefit Metrics

Establishing cost-benefits of prognostics is an important step in integrating the health management practices into fielded applications. Thus, metrics that measure economic viability of prognostics have started gaining in importance. Some of the most common cost-benefit metrics include

5.3.2.1 MTBF-to-MTBR Ratio

Statistics-based what?

This reliability statistics-based expresses the efficiency of a maintenance operation by measuring the ratio between the lengths of time a component is expected to last and the length of time for which it was used before it was replaced [17].

5.3.2.2 Life Cycle Cost

LCC is fundamentally the sum of acquisition cost and cost of operations. To assess the value of prognostics, LCC is compared with and without prognostics [18].

5.3.2.3 Return on Investment

In an ROI calculation, the difference between return and investment (the gain) is divided by the investment. It is one of the most commonly used metrics (not just in the context of prognostics) that assesses the benefits of deploying a PHM system.

5.3.2.4 Technical Value

The benefits achieved through accurate detection, fault isolation, and prediction of critical failure modes are weighed against the costs associated with false alarms, inaccurate diagnoses/prognoses, and resource requirements of implementing and operating specific techniques [19,20].

5.3.2.5 Total Value

Given the coverage a PHM system provides for multiple fault modes in a system, total value quantifies the usefulness of a PHM technology in a particular application. Total value is defined as the summation of the benefits prognostics provides over all the failure modes that it can diagnose or give a prognosis, minus the implementation cost, operation and maintenance cost, and consequential cost of incorrect assessments. This metric connects the algorithm performance in a PHM system to the management and operational performance.

5.3.3 Metrics for Computational Performance

Computational performance is one of the most closely related factors to actual implementation of a system. It provides a mechanism to negotiate between computational and time resources that are demanded and the required fidelity and accuracy of the PHM system. Most of the metrics that can be used to quantify computational performance come from theoretical computer science and computer engineering. These are not specific to prognostics and are just mentioned here for completeness. Computational performance metrics include computational complexity, CPU time, memory size, and data rate. Depending on the length of the prediction horizon in a prognostic application, data processing capabilities are of greater significance from the design and implementation point of view. All the above metrics help specify the hardware requirements or otherwise specify constraints within which a software must work and still satisfy algorithmic performance requirements.

5.3.4 Metrics for Reliability Analysis

Referring again to Table 5.1, reliability analysis metrics are chiefly used by operators, designers, and policy makers. Reliability analysis stems from statistical evidences aggregated from historical data. Failure rates and distributions are extracted from history data or experimental data, which are then used to make failure predictions for a system under test. While prognostics is the science of prediction based on condition and usage of the monitored UUT, the reliability analysis predicts failures based on expected outcome from the observed statistic over a population of the UUT. It has been the traditional way to use these metrics to assess the costs and the risks of using a system. Used correctly, reliability metrics are connected to system improvement due to prognostic performance.

Broadly classifying there are two types of reliability metrics as described below [21]:

5.3.4.1 Constant Rate Reliability Metrics

These are the most popular reliability metrics in the electronics industry as they represent a good approximation of the flat region of the reliability bathtub curve. Mean life metrics usually assume an exponential distribution, which makes them equivalent to constant rate metrics. These rates are typically measured from field data and are simple and intuitive to explain. Some common examples of these metrics are mean time between failure (MTBF), mean time to failure (MTTF), part return/repair rate, part replacement rate, mean time between service call (MTBSC), and mean time between maintenance action (MTBMA).

5.3.4.2 Probability of Success Metrics

When systems do not show constant failure rates, specifying mean times does not suffice. In such cases a better way is to specify probability of success or, in other words, the probability that a system performs a required function under stated condition for a stated period. Another way to specify probability of success is to measure the percentage of population that survives a specific duration. Therefore, these metrics are usually time dependent, i.e., the probability of success will depend on the length of the mission. These may be specified as the percentiles of the distributions. A common example used in mechanical systems domain is *Lx Life*, which specifies the number of hours after which at least *x*% of the population would have failed. Other metrics commonly used are failure free operating time, maintenance free operating time, mean mission duration, etc. More discussion on reliability-based metrics may be found in [22].

5.3.5 Metrics for Prognostics Algorithm Performance

Before using performance metrics, an a priori analysis should be conducted to identify the relevant factors in a given application and address them appropriately. A good set of metrics should accommodate all or most of these factors. In this context, the challenges surrounding prognostics (as compared with, say, diagnostics) should be discussed. It should also be noted that the metrics detailed in this chapter are continuously evolving as the field matures further.

5.3.5.1 Challenges

Prognostics requires special consideration in a number of areas. These include the acausality phenomenon, need for run-to-failure data, online performance evaluation, and expression of estimation uncertainty.

Acausality: An acausal system is defined as a system with outputs and internal states that depend on future input values. Prognostics has acausal properties. It requires input from future events (for instance knowledge about operational conditions and load profiles) to make accurate predictions. To accurately assess the performance (both accuracy or precision), one must also know the true EoL to compare with the predicted EoL estimates. In some cases, future operating conditions are well known. This is the case for example for stationary applications with constant operating conditions. However, in nonstationary applications and where knowledge about future events is not available, estimates may be derived based on past usage history, the expected mission profile, and predictions for future operating and environmental conditions that are not controllable (e.g., weather conditions). This however, adds uncertainty to the overall process and complicates prognostic performance evaluation.

Run-to-failure data from real applications: Assessing the correctness of prognostics benefits greatly from allowing the system to fail such that the prediction can be confirmed. For many systems, this is not feasible because it may be too expensive or because it negatively impacts system safety. However, if a corrective action (such as maintenance or repair) is taken, one has just removed the ability to assess how early the prediction was. This is sometimes referred to as the "paradox of prognostics."

Online performance evaluation: The aforementioned considerations lead to an argument in favor of controlled run-to-failure (RtF) experiments for the algorithm development phase. While this may allow offline performance evaluation, some issues remain: First, it is difficult to extend the results of offline conditions to a real-time scenario; second, an RtF experiment needs often times frequent disassemblies to gather ground truth data. This assembly-disassembly process creates variations in the system performance, and the EoL point shifts from what it may have been in the beginning of the experiment. Since actual EoL is observed only at the end there is no guarantee that a prediction made based on initial part of data will be very accurate. Whereas, this does not necessarily mean that prognostic algorithm is poorly trained, it is difficult to prove otherwise. Therefore, one must be careful while interpreting the performance assessment results. Third, even controlled subscale RtF experiments can

be very expensive and time-consuming, in particular, if one seeks to conduct statistically significant number of experiments for an exhaustive set of components and fault modes.

Uncertainty in prognostics: The quantification of prediction confidence is indispensable in prognostics. Consider a remaining life estimate of, say, 5 hours. If one knew that the confidence bounds for a given risk acceptance level are also 5 hours, then reactive action has to be taken immediately. If, however, the confidence bounds are at ±1 hour, then a completely different set of action can be taken. Without such information any prognostic estimate is of limited use and cannot be incorporated in mission critical applications [6]. Uncertainties arise from various sources in a PHM system [23–25]. Some of these sources include

- Model uncertainties (errors in the representation and parameters of both the system model and fault propagation model),
- Measurement uncertainties (these arise from sensor noise, ability of sensor to detect and disambiguate between various fault modes, loss of information due to data preprocessing, approximations, and simplifications),
- Operating environment uncertainties,
- Future load profile uncertainties (arising from unforeseen future and variability in usage history data),
- Input data uncertainties (estimate of initial state of the system, variability in material properties, manufacturing variability), etc.

Assessing the levels and characteristics of uncertainties arising from each of these sources is often times not a trivial task. It is even more difficult to determine how these uncertainties combine at different stages of the prognostic process and propagate through the possibly complex and nonlinear system. On top of that, statistical properties may not follow any known parametric distributions, thus complicating analytical solutions.

Owing to all of these challenges, uncertainty representation and management has become an active area of research in the field of PHM [25–29]. Methods for prognostic performance evaluation must then be able to incorporate various expressions of uncertainties.

Performance metrics for prognostics can be classified into accuracy, precision, and robustness. We use the working definition for accuracy as

those that assess the degree of closeness of predictions to the actual failure time. Precision is defined as the spread of predictions performed at the same time. Robustness is defined as the sensitivity of the predictions with changes of algorithm parameter variations or external disturbances. There are a large number of prognostic performance metrics that have been used. However, as discussed earlier, most of these metrics do not take into consideration the particular challenges of prognostics. Hence, we feature here only a subset of general metrics especially suitable for prognostics. For a comprehensive list of performance metrics, the reader is referred to [9].

5.3.6 Error-Based Metrics

Many metrics are based on the assessment of the error, i.e., the deviation of the actual output from the target. One example of such an error metric is the average scale independent error. This metric provides an exponential weight of the errors in RUL predictions and averages over several UUTs [20,30]:

$$A(i) = \frac{1}{L}\sum_{l=1}^{L} \exp\left\{-\frac{\left|\Delta^{l}(i)\right|}{D_0}\right\} \tag{5.1}$$

where Δ is the error and D_0 is a normalizing constant whose value depends on the magnitudes in the application. The range of $A(i)$ varies between 0 and 1, where 1 represents perfect score. Other error-based metrics include root mean squared error (RMSPE) and mean absolute percentage error (MAPE).

5.3.6.1 FP, FN, and ROC

FP and FN are at heart also error-based metrics, but they deserve special consideration. A common way to assess performance is to treat predictions as dichotomous forecasts by means of categorizing them into false-positives (FP), false-negatives (FN), true-positives (TP), and true-negatives (TN) [10]. FP assesses unacceptable early predictions and FN assesses unacceptable late predictions at specified time instances. User must set acceptable ranges (t_{FN} and t_{FP}) for prediction. Early predictions result in excessive lead time, which may lead to unnecessary corrections. Also note that, a prediction that is late more than a critical threshold time units (t_c) is equivalent to not making any prediction and having the failure occurring. Mathematically, FP is defined as

$$\mathrm{FP}\left(r_*^l(i)\right)=\begin{cases}1 & \text{if } \Delta^l(i)>t_{FP}\\ 0 & \text{otherwise}\end{cases} \tag{5.2}$$

where t_{FP} is the user-defined acceptable early prediction and FN is defined as

$$\mathrm{FN}\left(r_*^l(i)\right)=\begin{cases}1 & \text{if } -\Delta^l(i)>t_{FN}\\ 0 & \text{otherwise}\end{cases} \tag{5.3}$$

where t_{FN} is the user-defined acceptable late prediction.

FP and FN both can vary between values is 0 and 1, where 1 represents perfect score. FP and FN can then be compiled into ROC curve. The ROC allows to assess the tradeoff between FP and FN [31,32] in a comprehensive fashion by plotting (1 – FNs) over the FPs (see Figure 5.1). The ideal curve would have zero FPs and zero FNs, but such a curve cannot realistically be achieved for real-world problems. Use of time-dependent ROC has been suggested that depicts ROC obtained for forecasts made for different time horizons. Also, each point on the ROC curve may be associated with a point wise fixed width confidence bounds to indicate confidence in predictions. Tuning the prognostic algorithm such that a

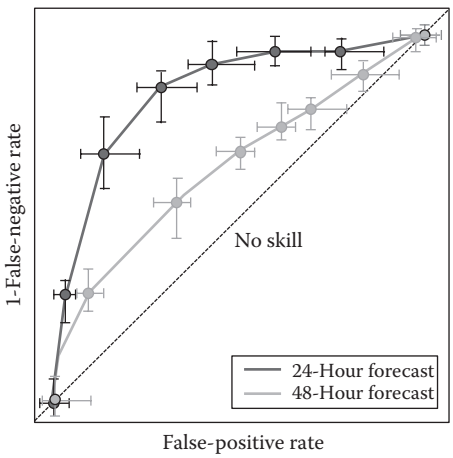

FIGURE 5.1 ROC curve.

ROC can be generated may prove difficult in practice (e.g., due to lack of data or lack of tuning "parameters").

5.3.6.2 Spread-Based Metrics

Spread-based metrics measure the dispersion/spread of the error. The most basic spread-based metric is the sample deviation, which considers the error dispersion with respect to the error sample mean [20,33].

$$S(i)=\sqrt{\frac{\sum_{l=1}^{n}\left(\Delta^{l}(i)-M\right)^{2}}{n-1}} \tag{5.4}$$

where M is the sample mean of the error. This metric is restricted to the assumption of normal distribution of the error. It is, therefore, recommended to carry out a visual inspection of the error plots. SD can vary between 0 and ∞, where the perfect score is 0. Other spread-based metrics include MAD and MdAD from the sample median.

5.3.6.3 Anomaly Correlation Coefficient

This metric is used to measure correspondence or phase difference between prediction and observations, subtracting out the historical mean at each point and is frequently used to verify output from numerical prediction models [7]. Anomaly correlation coefficient (ACC) is not sensitive to error or bias, so a good anomaly correlation does not guarantee accurate predictions. In the PHM context, ACC computed over a few time-steps after t_P can be used to modify long-term predictions. However, the method requires computing a baseline from history data, which may be difficult to come by.

Mathematically, ACC can be represented as follows:

$$\mathrm{ACC}=\frac{\sum\left(\pi^{l}(i|j)-z_{\#}(i)\right)\left(z_{*}(i)-z_{\#}(i)\right)}{\sqrt{\sum\left(\pi^{l}(i|j)-z_{\#}(i)\right)^{2}\sum\left(z_{*}(i)-z_{\#}(i)\right)^{2}}}, \tag{5.5}$$

where $z_*(i)$ is a prediction variable (e.g., $f^{l}_{*_n}(i)$ or $h^{l}_{*}(i)$) and $z_{\#}(i)$ is the corresponding history data value. ACC can vary between –1 and 1, where 1 represents perfect score.

Thus, ACC averages the absolute percentage errors in the predictions of multiple UUTs at the same prediction horizon. The percentage is computed based on the mean value of the prediction and ground truth. This prevents the percentage error from being too large for the cases where the ground truth is close to 0. This metric is computed at a particular time and does not capture performance variation with time.

5.3.6.4 Prognostic Horizon

The prognostic horizon (PH) can be formally defined as the difference between the time index i when the predictions first meet the specified performance criteria (based on data accumulated until time index i) and the time index for EoL. PH can be considered as a robustness metric. The basic notion behind the metric is that a longer PH implies more time to act based on a prediction that has some credibility. The performance requirement is specified in terms of an allowable error bound (α) around the true EoL where the choice of α depends on the estimate of time required to take a corrective action. PHs are typically determined offline during the validation phase for an algorithm-application pairing. PH performance is then used as a guideline for algorithm deployment where actual EoL is not known in advance.

$$\text{PH} = \text{EoL} - i \tag{5.6}$$

where $i = \min\left\{ j \,|\, \left(j \in l\right) \wedge \left((r_* - \alpha.\text{EoL}) \le r^l(j) \le (r_* + \alpha \cdot \text{EoL})\right)\right\}$.

PH output is a score that is characterized by both the length of remaining life of a system and the time scales in the problem at hand. As shown in Figure 5.2, the desired level of accuracy with respect to the EoL ground truth is specified as ±α-bounds. A remaining life estimate within those bounds has sufficient utility to a user (it is not too far off from the target to be actionable). The PH for an algorithm is declared as soon the corresponding predictions enter the α-bounds. RUL values can be superimposed for various algorithms, thus providing an easy aid in their comparison. As evident from Figure 5.2, algorithm A1 has a longer PH than algorithm A2.

5.3.6.5 α-λ Performance

α-λ metric quantifies the prediction quality by determining whether the prediction falls within specified limits at particular times with respect to a performance measure. The evaluation times may be specified either as

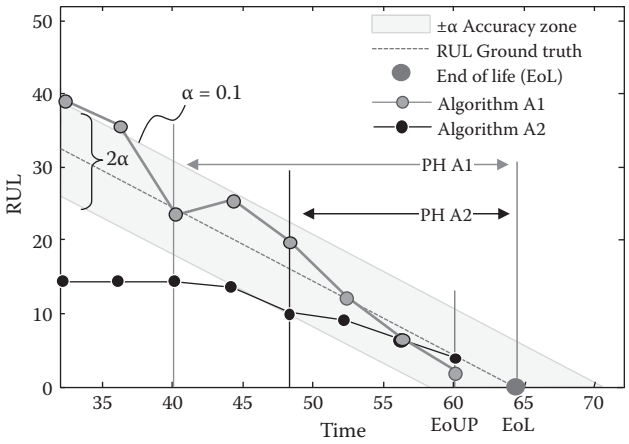

FIGURE 5.2 Prognostic horizon.

a ratio to the total remaining life from the point the first prediction is made or it may be specified as a given absolute time interval before EoL is reached. α-λ Performance could be expressed as α-λ accuracy, α-λ precision, or α-λ robustness metric. In the discussion below, we delineate α-λ performance without loss of generality as an accuracy performance measure.

Here we define α-λ accuracy as the prediction accuracy to be within α % of the actual RUL at specific time instance expressed as a fraction of time between the point when an algorithm starts predicting and the actual failure (Figure 5.3). Consider an example case where this metric determines whether a prediction falls within 20% accuracy (i.e., α = 0.2) halfway to failure from the time the first prediction is made, (i.e., λ = 0.5). The α-λ accuracy metric is defines as

$$\alpha\text{-}\lambda \text{ Accuracy} = \begin{cases} 1 & \text{if } \left(1-\alpha\right)\cdot r_*\left(i\right) \le r^l\left(i_\lambda\right) \le \left(1+\alpha\right)\cdot r_*\left(i\right) \\ 0 & \text{otherwise} \end{cases} \quad (5.7)$$

where α is the accuracy modifier and λ is the time window modifier such that $t_\lambda = t_P + \lambda(t_{EoL} - t_P)$.

There are two Saxena et al. 2009. Please check which should be cited here.

For illustrating the usage of this performance measure, several prediction algorithms employed in [34–37] are compared in Figure 5.4. Here

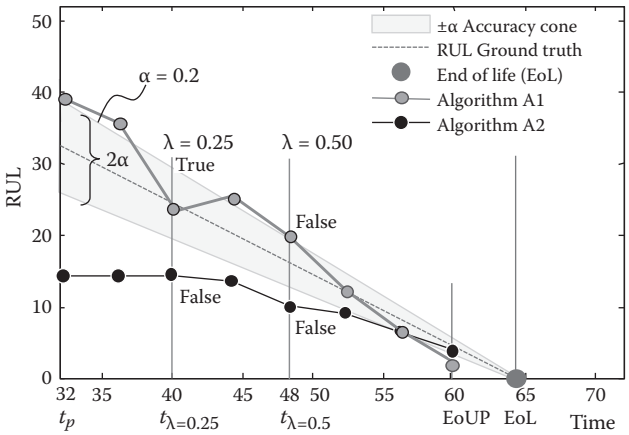

FIGURE 5.3 α-λ Performance.

this metric evaluates whether predictions made by various algorithms lie within 10% error when evaluated at halfway to the EoL.

5.3.6.6 Relative Accuracy

Relative accuracy (RA) is defined as a measure of error in RUL prediction relative to the actual RUL.

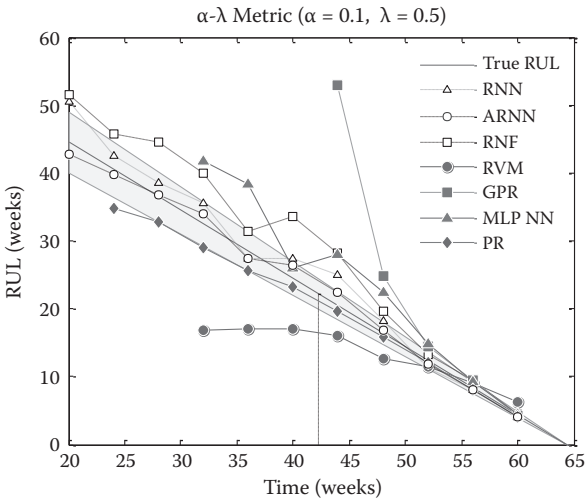

FIGURE 5.4 α-λ Performance comparison.

$$\mathrm{RA}_\lambda = 1 - \frac{\left| r_* \left(i_\lambda\right) - \left\langle r^l \left(i_\lambda\right)\right\rangle \right|}{r_* \left(i_\lambda\right)} \tag{5.8}$$

where λ is the time window modifier such that $t_\lambda = t_P + \lambda(t_{EoL} - t_P)$, l is the index for lth UUT, $r_*(i_\lambda)$ is the ground truth RUL at time index i_λ, and $\langle r(i_\lambda)\rangle$ is an appropriate point estimate of the predicted RUL distribution at time index i_λ.

5.3.6.7 Cumulative Relative Accuracy

RA conveys information at a specific time. To estimate the general behavior of the algorithm, RA can be evaluated at multiple time instances to provide an aggregate accuracy level or the cumulative relative accuracy (CRA).

$$\mathrm{CRA}_\lambda = \frac{1}{\left|\ell_\lambda\right|} \sum_{i \in \ell_\lambda} w\left(r(i)\right) \mathrm{RA}_\lambda \tag{5.9}$$

where $w(r(i))$ is a weight factor as a function of RUL at all time indices and l_λ is the set of all time indexes when a prediction was made.

It may be desirable to give more weight to RA evaluated at times closer to EoL since good performance close to EoL is important for condition-based decision making. Therefore, one would expect that λ is chosen in a meaningful fashion, e.g., the time required to apply a corrective action. RA evaluated at λ = 0.5 indicates the time when a system is expected to have consumed half of its remaining life. Alternatively, RA could be evaluated at time instances where the damage magnitude has reached 50% of the failure threshold. This metric is also useful in comparing different algorithms for a given λ to get an idea on how well a particular algorithm does at significant times.

5.3.6.8 Convergence

Convergence expresses the rate at which any metric (M), such as accuracy or precision, improves with time. The error of different algorithm metric evaluation is connected into a curve. Convergence is then defined as the distance between the origin and the centroid of the area under the curve for a given metric.

$$C_M = \sqrt{(x_c - t_P)^2 + y_c^2}, \tag{5.10}$$

where C_M is the Euclidean distance between the center of mass (x_c, y_c) and $(t_P, 0)$, $M(i)$ is a nonnegative prediction accuracy or precision metric with a time varying value, (x_c, y_c) is the center of mass of the area under the curve $M(i)$ between t_P and t_{EoUP}, defined as following

$$x_c = \frac{\frac{1}{2}\sum_{i=P}^{EoUP}(t_{i+1}^2 - t_i^2)M(i)}{\sum_{i=P}^{EoUP}(t_{i+1} - t_i)M(i)},$$

$$y_c = \frac{\frac{1}{2}\sum_{i=P}^{EoUP}(t_{i+1} - t_i)M(i)^2}{\sum_{i=P}^{EoUP}(t_{i+1} - t_i)M(i)} \tag{5.11}$$

where EoUP (end of useful predictions) is the time index for last useful prediction made. Alternatively, one may use EoP, but EoUP makes sure that performance is evaluated only based on those predictions that are useful from a practical view point since any prediction made after EoUP does not leave enough time to carry out any corrective measure [36,37].

There are two Saxena et al. 2009. Please check which should be cited here.

As stated earlier, convergence banks on the implicit assumption that algorithm performance should improve with time. For illustration of the concept, consider three cases that converge at different rates in Figure 5.5. Lower distance implies a faster convergence.

5.3.6.9 Robustness

A robustness metric has the task of quantifying the sensitivity of an algorithm with respect to its parameters, such as those found in expressing prior distribution, initial conditions, and training data size. Confidence bounds of a robust algorithm are not expected to change much with variations of algorithm parameters. Mathematically, the robustness metric R_b can be defined as

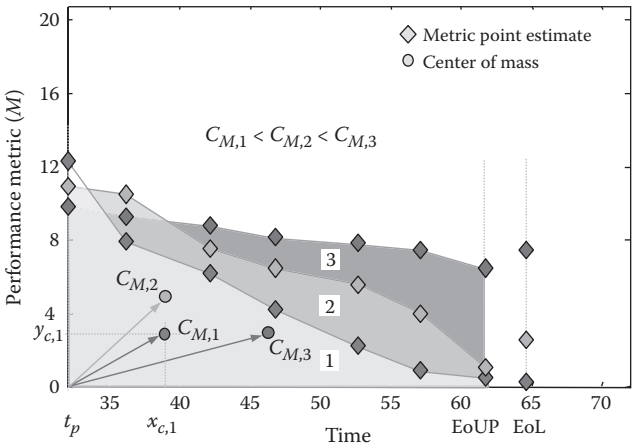

FIGURE 5.5 Convergence metric.

$$R_b = \frac{\int_{Xmean-\eta}^{Xmean+\eta} f(x)dx}{4\eta\delta} \tag{5.12}$$

where x is the investigated algorithm parameter and $f(x)$ is the confidence bound variation function with respect to x.

The assessment of algorithm robustness is of high value in particular since most of the time an accurate prior is difficult to obtain with limited data source and extensive experiments on the actual engineering system are often prohibitive due to time and cost constraints [38].

5.3.6.10 RUL Online Precision Index

This index quantifies the length of 95% confidence bounds relative to the predicted RUL at any given time instant [39]. The index is normalized between 0 and 1. It can be used as an online performance metric to ensure if I_1 remains close to 1 as system deteriorates, i.e., EoL approaches

$$I_1(i) = e^{-\left(\frac{\sup\{CI(i)\}-\inf\{CI(i)\}}{r(i)}\right)} \tag{5.13}$$

where $0 < I_1 \leq 1$, $\forall i \in [1, \langle \text{EoL} \rangle]$, $i \in I^+$

5.3.7 Incorporating Uncertainty Estimates

Prognostics algorithms typically involve estimating the probability distribution function (PDF) of the EoL and RUL, rather than single point predictions, which enables them to handle uncertainties arising from various sources such as noise, loading conditions, and so on. They also allow for propagation of uncertainties for subsequent predictions [40]. Thus, it is necessary to ensure that prognostic performance metrics include these factors. The most common form of assessing a PDF output is through estimates of mean and variance of the distribution owing to their simplicity and easy interpretation [34]. However, in reality, these distributions are rarely smooth or symmetric and hence mean and variance are not robust enough to evaluate the performance. A combination of mean as the measure of location and quartiles or interquartile range (IQR) as a measure of spread can provide better estimates of the distribution [33].

The metrics shown in the previous sections do not explicitly accommodate for uncertainty estimating capability of the prognostic algorithms. However, a fairly straightforward way to do so is to specify an allowable error bound for a given metric. This error bound could be asymmetric as shown in Figure 5.6. In case of prognostics, typically, a wider error margin to the "left" of the prediction (that is, an early prediction) may be preferred because early predictions have lower cost and safety consequences than late ones.

These concepts can be analytically incorporated into the metrics by calculating the probability mass of a prediction falling within the specified

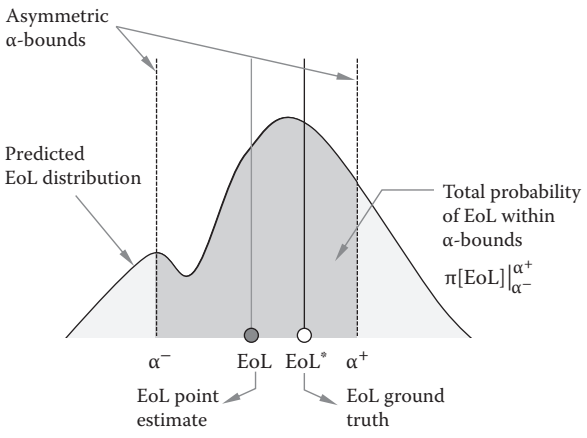

FIGURE 5.6 Concepts for incorporating uncertainties [4,5] (CC 3.0).

There are two Saxena et al. 2010. Please check which should be cited here.

α-bounds. As an illustrative example, consider again the α-λ accuracy. The α-bounds are expressed as a percentage of actual RUL $r(i_\lambda)$ at t_λ.

$$\alpha\text{-}\lambda\ \text{Accuracy} = \begin{cases} 1 & \text{if} \quad \pi\left[r(i_\lambda)\right]\Big|_{-\alpha}^{+\alpha} \geq \beta \\ 0 & \text{otherwise} \end{cases} \tag{5.14}$$

where $r(i_\lambda)$ is the predicted RUL at time index i_λ, $\pi\left[r(i_\lambda)\right]\Big|_{\alpha^-}^{\alpha^+}$ is the probability mass of the prediction PDF within the α-bounds that are given by $\alpha^+ = r_*(i_\lambda) + \alpha \cdot r(i_\lambda)$ and $\alpha^- = r_*(i_\lambda) - \alpha \cdot r(i_\lambda)$.

With α = 0.1 and λ = 0.5, the criteria for matching the metric requirement is determined by assessing the intersection of the uncertainty with the α-cone as shown in Figure 5.7. Note that there may be no prediction assessed at time t_λ for a given λ and the corresponding $i_\lambda \notin p$ because the set of time indexes (p) where a prediction is made is determined by the frequency of prediction step in a prognostic algorithm. In such cases, one can make choose λ′ closest to λ such that $i_{\lambda'} \in p$. To illustrate the application of α-λ accuracy further, refer to Figure 5.8, where the performance of a Recurrent Neural Network algorithm is plotted for every time instant when a prediction was made. Figure 5.8 indicates at any point with either

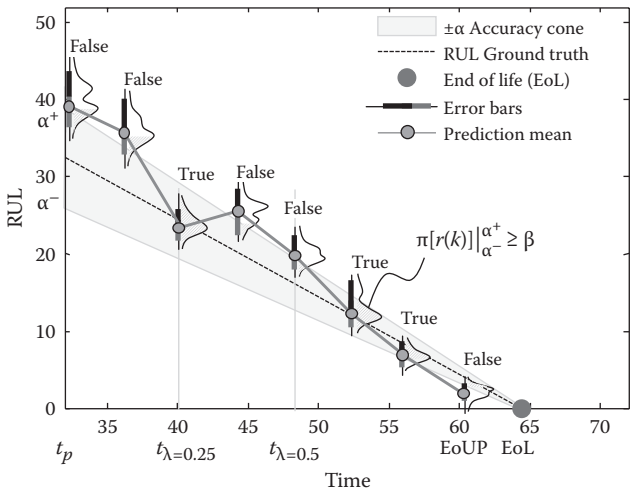

FIGURE 5.7 α-λ Accuracy with the accuracy cone shrinking with time on RUL versus time plot.

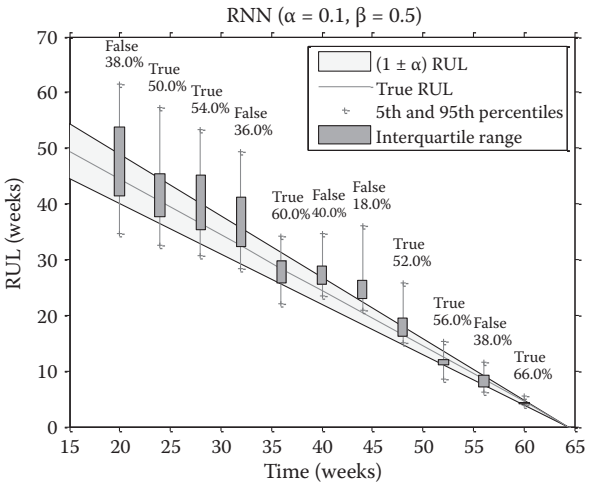

FIGURE 5.8 α-λ Accuracy for a particular algorithm with distribution information [35] (CC 3.0).

"true" or "false" (written above the upper quartiles) whether α-λ accuracy metric is satisfied or not based on β-criterion.

5.3.8 Guidelines for Applying Prognostics Metrics

Given the structure of the prognostics metrics described in this chapter, one can observe a progression in the manner how these metrics characterize the algorithm performance. The first metric, PH, identifies whether an algorithm predicts within a specified error limits around the actual EoL and, if it does, how much time it allows for any corrective action. Therefore, if an algorithm does not allow a sufficient PH it may not be meaningful to continue on computing other metrics. On the other hand, if an algorithm passes the PH test, the next metric, α-λ performance, identifies whether the algorithm performs within desired error margins of the actual RUL at any given time instant that may be of interest to in a particular application. This is a more stringent requirement of staying within a converging cone of the error margins as a system nears EoL. If this criterion is also met, the next step is to quantify the accuracy levels relative to the actual RUL. This is accomplished by the metrics RA and CRA. These metrics assume that prognostic performance improves as more information becomes available with time, and hence, by design, an algorithm will satisfy these metrics criteria if it converges to true RULs. Therefore, the

fourth metric, convergence, quantifies how fast the algorithm converges if it does satisfy all previous metrics. These metrics can be considered as a hierarchical test that provides several levels of comparison among different algorithms in addition to the specific information these metrics individually provide regarding algorithm performance. Of course, the use of other metrics such as robustness stands by itself to assess sensitivity of any of these or even other metrics with respect to a key system parameter.

5.3.8.1 Guidelines on Choosing Performance Parameters

Time critical nature of prognostic application resulted in metrics for which the performance evolves with time and needs to be tracked. This required several special parameters that must be specified to define time criticality (λ), confidence level (β), or acceptable error bounds (α). The choice of α depends on the estimate of time required to take a corrective action. Depending on the situation this corrective action may correspond to performing maintenance (manufacturing plants) or bringing the system to a safe operating mode (operations in a combat zone). Adjustments to these parameters may translate into significant changes in the cost-benefit-risk equation in a process. Therefore, it is suggested that these parameters be chosen carefully to clearly specify prognostic requirements [4,5]. Requirements engineering is a discipline that provides guidelines to obtain these requirements in a systematic manner. For instance, in a safety critical military application, first, a failure modes affects and criticality analysis (FMECA) or hazard and operability analysis (HAZOP) must be conducted to identify most critical failures. Then, based on available sensors, measurement quality, noise levels, etc. desired confidence levels must be derived. For safety critical systems, a more conservative failure threshold may be chosen, while for commercial applications a less conservative but more cost effective threshold is preferred. It must be noted that the choice of metrics and performance specifications is an iterative process that negotiates between user requirements and constraints originating from performance needs, available resources, established maturity level of PHM, and time criticality for that application.

There are two Saxena et al. 2010. Please check which should be cited here.

5.3.8.2 Guidelines for Dealing with Uncertainties

A prognostic system models a stochastic process, and hence, the behavior observed from a particular run (single realization of the stochastic process) does not represent the complete behavior of the predicted

trajectories. Assuming that all measures practically possible for uncertainty reduction have been taken during the algorithm development phase, such observations should be treated only as isolated realization of the process. A level of confidence or probability of occurrence should be attached to such predictions. Otherwise, multiple trajectories should be aggregated from several runs to achieve statistical significance and more sophisticated stochastic analyses may be carried out. Another aspect dealing with uncertainties is related to prognostic algorithm output. Different algorithms represent uncertainties in different ways. Some specify parametric distribution and other as nonparametric ones. Furthermore, some result in a closed form analytical equation for these distributions and other only result in discretized histograms. It is very important to carefully treat these distributions and not lose critical information by approximating these by known simpler forms such as normal distribution or by computing their statistical moments [4,5,36,37]. A common practice has been to compute mean and variance for all types of distributions whereas they may not be very meaningful for nonnormal distributions. Use of more robust estimators such as median, L-estimator, or M-estimator for expressing central tendency and IQR, MAD, or MdAD for expressing the spread is suggested [41].

There are two Saxena et al. 2009. Please check which should be cited here.

There are two Saxena et al. 2010. Please check which should be cited here.

5.3.8.3 Guidelines to Resolve Ambiguities

In practice, there can be several situations where the definitions discussed above result in ambiguity. Some of such situations are very briefly discussed here with suggested resolutions.

While applying the PH metric, a common situation encountered is when the RUL trajectory jumps out of the ±α accuracy bounds temporarily. Situations like this result in multiple time indexes where RUL trajectory enters the accuracy zone to satisfy the metric criteria. A simple and conservative approach to deal with this situation is to declare a PH at the latest time instant the predictions enter accuracy zone and never comes out thereafter. Another option is to use the original PH definition and further evaluate other metrics to determine whether the algorithm satisfies all other requirements. As discussed by Saxena et al. [36,37], situations such as these can occur due to a variety of reasons.

There are two Saxena et al. 2009. Please check which should be cited here.

Inadequate system model: Real systems often exhibit inherent transients at different stages during their life cycles. These transients get reflected as deviations in computed RUL estimates from the true value if the underlying model assumed for the system does not account for these behaviors. In

such cases, one must step back and refine the respective models to incorporate such dynamics.

Operational transients: Another source of such behaviors can be due to sudden changes in operational profiles under which a system is operating. Prognostic algorithms may show a time lag in adapting to such changes and hence resulting in temporary deviation from the real values. Therefore, whenever inconsistent behavior of PH metric is observed, one must identify the root cause of it and accordingly interpret the results. The situations discussed here are more common typically towards the end when a system nears EoL. This is because in most cases the fault evolution dynamics are too fast and complex to model or learn from data as the system nears EoL. Therefore, RUL curve deviates from the error band near t_{EoL}. To determine whether such deviations are critical for postprognostic decision making, the concept of t_{EoUP} or EoUP is introduced. This index represents the minimum allowable PH that is required to take a corrective measure. Any predictions made beyond EoUP are of little or no use from a practical viewpoint.

5.4 SUMMARY

This chapter presents several performance metrics for offline evaluation of prognostics algorithms. A brief overview of different methods employed for performance evaluation is also included. Because metrics developed in the context of forecasting differ from prognostics in the systems health management context and to account for the additional considerations, metrics specialized for prognostics (but not necessarily for the application) are needed. These metrics were introduced, and their use was illustrated with recommendations.

ACKNOWLEDGMENTS

The authors would like to express their gratitude to colleagues at the Prognostic Center of Excellence (NASA Ames Research Center) and external partners at Impact Technologies, Georgia Tech, and Clarkson University for participating in research discussions, evaluating metrics in their respective applications, and providing a valuable feedback. This work was funded by NASA Aviation Safety Program-IVHM Project.

REFERENCES

[1] M. Schwabacher. *A Survey of Data Driven Prognostics.* AIAA Infotech@ Aerospace Conference, Arlington, VA, 2005.

[2] M. Schwabacher, K. Goebel. *A Survey of Artificial Intelligence for Prognostics.* AAAI Fall Symposium, Arlington, VA, 2007.
[3] Condition Monitoring and Diagnostics of Machines—Prognostics part 1: General Guidelines, ISO/IEC Directives Part 2 C.F.R., 2004.
[4] A. Saxena, J. Celaya, B. Saha, S. Saha, K. Goebel. Metrics for offline evaluation of prognostic performance. *International Journal of Prognostics and Health Management,* 1(1):21, 2010.
[5] A. Saxena, J. Celaya, S. Saha, B. Saha, I. Roychoudhury, K. Goebel. *Requirements Specification for Prognostics Performance—An Overview*. AIAA Infotech @ Aerospace, Atlanta GA, 2010.
[6] S. Uckun, K. Goebel, P.J.F. Lucas. Standardizing research methods for prognostics. In *International Conference on Prognostics and Health Management (PHM08)*, Denver, CO. 2008.
[7] S.J. Engel. Prognosis requirements and V&V: Panel discussion on PHM capabilities: Verification, validation, and certification issues. In *International Conference on Prognostics and Health Management (PHM08)*, Denver, CO, 2008.
[8] S.J. Engel, B. Gilmartin, K. Bongort, A. Hess. Prognostics, the real issues involved with predicting life remaining. In *IEEE Aerospace Conference*, Big Sky, MT, 2000.
[9] A. Saxena, J. Celaya, E. Balaban, K. Goebel, B. Saha, S. Saha. Metrics for evaluating performance of prognostics techniques. In *1st International Conference on Prognostics and Health Management (PHM08)*, Denver, CO, 2008.
[10] K. Goebel, P.P. Bonissone. Prognostic information fusion for constant load systems. In *7th Annual Conference on Information Fusion*, 2005.
[11] P. Sandborn. A decision support model for determining the applicability of Prognostic Health Management (PHM) approaches to electronic systems. In *Reliability and Maintainability Symposium (RAMS)*, Arlington, VA, 2005.
[12] C.S. Byington, M.J. Roemer, P.W. Kalgren. Verification and validation of diagnostic/prognostic algorithms. In *Machinery Failure Prevention Technology Conference (MFPT 59)*, Virginia Beach, VA, 2005.
[13] G.J. Kacprzynski, M. Gumina, M.J. Roemer, D.E. Caguiat, T.R. Galie, J.J. McGroarty. A prognostic modeling approach for predicting recurring maintenance for shipboard propulsion system. In *55th Meeting of the Society for Machinery Failure Prevention Technology*, Virginia Beach, VA, 2001.
[14] J.J. Luna. Metrics, models and scenarios for evaluating PHM effects on logistics support. *Annual Conference of the Prognostics and Health Management Society*, San Diego, CA, 2009.
[15] J.B. Coble, J.W. Hines. Prognostic algorithm categorization with PHM challenge application. In *1st International Conference on Prognostics and Health Management (PHM08)*, Denver, CO, 2008.
[16] K.R. Wheeler, T. Kurtoglu, S. Poll. A survey of health management user objectives related to diagnostic and prognostic metrics. In *ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE)*, San Diego, CA, 2009.

[17] C. Teal, B. Larsen. Technology update II: Wire systems diagnostics and prognostics. In *Digital Avionics Systems Conference*, 2003.
[18] C. Wilkinson, D. Humphrey, B. Vermeire, J. Houston. Prognostic and health management for avionics. In *IEEE Aerospace Conference*, Big Sky, MT, 2004.
[19] J.E. Dzakowic, G.S. Valentine. Advanced techniques for the verification and validation of prognostics and health management capabilities. *In Machinery Failure Prevention Technologies Conference (MFPT 60)*, Virginia Beach, VA, 2006.
[20] G. Vachtsevanos, F.L. Lewis, M. Roemer, A. Hess, B. Wu. Intelligent fault diagnosis and prognosis for engineering systems (1st edition). John Wiley & Sons, Hoboken, New Jersey, 2006.
[21] A.P. Wood. *Reliability-metric varieties and their relationships.* In *Reliability and Maintainability Symposium, 2001*, Philadelphia, 2001.
[22] IEEE-Standard 1413. IEEE Guide for Selecting and Using Reliability Predictions Based on IEEE 1413™ *1413.1TM* (pp. 97): IEEE Standards Coordinating Committee 37, 2002.
[23] A. Coppe, R.T. Haftka, N. Kim, F. Yuan. Reducing uncertainty in damage growth properties by structural health monitoring. In *Annual Conference of the Prognostics and Health Management Society (PHM09)* San Diego, CA, 2009.
[24] D. Hastings, H. McManus. A framework for understanding uncertainty and its mitigation and exploitation in complex systems. In *Engineering Systems Symposium MIT*, Cambridge MA, 2004.
[25] M. Orchard, G. Kacprzynski, K. Goebel, B. Sahaand, G. Vachtsevanos. Advances in uncertainty representation and management for particle filtering applied to prognostics. In *International Conference on Prognostics and Health Management (PHM08)*, Denver, CO, 2008.
[26] R. DeNeufville. Uncertainty management for engineering systems planning and design. In *Engineering Systems Symposium MIT*, Cambridge, MA, 2004.
[27] K.-C. Ng, B. Abramson. Uncertainty management in expert systems. *IEEE Expert Systems,* 5:20, 1990.
[28] S. Sankararaman, Y. Ling, C. Shantz, S. Mahadevan. Uncertainty quantification in fatigue damage prognosis. In *Annual Conference of the Prognostics and Health Management Society (PHM09)*, San Diego, CA, 2009.
[29] L. Tang, G. Kacprzynski, K. Goebel, G. Vachtsevanos. Methodologies for uncertainty management in prognostics. In *IEEE Aerospace Conference*, Big Sky, MT, 2009.
[30] R.J. Hyndman, A.B. Koehler. Another look at measures of forecast accuracy. *International Journal of Forecasting*, Elsevier, 22(4):679–688, 2006.
[31] B. Ebert. Forecast Verification—Issues, Methods and FAQ, from http://www.bom.gov.au/bmrc/wefor/staff/eee/verif/verif_web_page.html, 2007.
[32] T.N. Palmer, A. Alessandri, U. Andersen, et al. Development of a european multimodel system for seasonal-to-interannual prediction (Demeter), *Bulletin of the American Meteorological Society*, 85:853–872, 2004.

[33] D.C. Hoaglin, F. Mosteller, J.W. Tukey, editors. Understanding robust and exploratory data analysis. John Wiley & Sons, 1983.
[34] K. Goebel, B. Saha, A. Saxena, J. Celaya, J.P. Christopherson. Prognostics in battery health management. *IEEE Instrumentation and Measurement Magazine,* 11:33–40, 2008.
[35] J. Liu, A. Saxena, K. Goebel, B. Saha, W. Wang. An adaptive recurrent neural network for remaining useful life prediction of lithium-ion batteries. In *2nd Annual Conference of the Prognostics and Health Management Society (PHM10)*, Portland OR, 2010.
[36] A. Saxena, J. Celaya, B. Saha, S. Saha, K. Goebel. Evaluating algorithmic performance metrics tailored for prognostics. In *IEEE Aerospace Conference*, Big Sky, MT, 2009.
[37] A. Saxena, J. Celaya, B. Saha, S. Saha, K. Goebel. On applying the prognostics performance metrics. In *Annual Conference of the Prognostics and Health Management Society (PHM09)*, San Diego, CA, 2009.
[38] X. Guan, R. Jha, Y. Liu, A. Saxena, J. Celaya, K. Goebel. Comparison of two probabilistic fatigue damage assessment approaches using prognostic performance metrics. *International Journal of Prognostics and Health Management*, 1–13, 2010.
[39] M.E. Orchard, L. Tang, K. Goebel, G. Vachtsevanos. A novel RSPF approach to prediction of high-risk, low-probability failure events. In *Annual Conference of the Prognostics and Health Management Society (PHM09)*, San Diego, CA, 2009.
[40] M.E. Orchard, G. Vachtsevanos. A particle-filtering approach for on-line fault diagnosis and failure prognosis. *Transactions of the Institute of Measurement and Control,* 31(3–4):221–246, 2009.
[41] J.L. Devore. *Probability and Statistics for Engineering and the Sciences* (6th edition). Thomson, 2004.